REMARKS

This Amendment is filed in response to the Office Action dated April 20, 2006. All objections and rejections are respectfully traversed.

Claims 1-41 are in the case.

Claims 30-41 were added to better claim the invention.

Claims 1-11 and 27-29 were amended to better claim the invention.

At paragraph 1 of the Office Action claims 1-20 and 27-29 were rejected under 35 U.S.C. § 101 as being unpatentable subject matter. The Court of Customs and Patent Appeals states:

> Practical utility is a shorthand way of attributing "real world" value to claimed subject matter. In other words, one skilled in the art can use a claimed discovery in a manner which provides some ***immediate benefit*** to the public.
> *Nelson v. Bowler*, 626 F.2d 853, 856, 206 USPQ 881, 883 (CCPA 1980).

Under heavy loads involving large files, "a file system utilizing CPs typically [halt] or otherwise suspend write operations...causing some application programs executing on the client to generate error messages or suffer failures due to timeout conditions." (page 5, lines 10-30). Applicant's claimed invention in claims 1-20 and 27-29 remedy this situation by allowing write allocation during CP resulting in an ***immediate benefit*** to the client of decreased latency caused by "a large number of incoming write operations that may be queued and suspended while the CP write allocation operation is performed." (page 5, lines 10-30).

At paragraph 2 of the Office Action claims 1-29 were rejected under 35 U.S.C. § 102(e) as being unpatentable in view of Koseki et al. U.S. Patent No. 6,732,124 issued on May 4, 2004, (hereinafter "Koseki").

The present invention, as set forth in amended representative claim 1 comprises:

> 1. A computer implemented method for managing data to be written to a file served by a storage system while the file is undergoing a write allocation procedure, the method comprising the steps of:
>
> receiving a write operation comprising data to be written to the file;
>
> associating the received data with a buffer data control structure associated with the file; and
>
> marking the buffer data control structure associated with the file as being ***dirty for a next consistency point.***

Koseki teaches a method for "managing the allocation of metadata volume resources on the basis of allocation management data." The basis for using an "allocation-dirty flag" is to determine whether "one or more allocating operations have been done to the bitmap block." Simply stated, a "'dirty' [state of a bitmap block] implies that information on the memory has been modified, but the modification has not yet been reflected in the disk storage...where as [a] 'clean' state of a bitmap block denotes the block has not experienced any allocation or deallocation operations." [col. 19, lines 35 to col. 21, lines 11].

Applicant respectfully urges that Koseki does not show Applicant's claimed novel use of a buffer data control structure to indicate if data is "***dirty for a next consistency point.***"

Koseki makes use of its "dirty" flags to indicate whether "one or more allocating operations have been done to the bitmap block." Koseki uses this technique for "efficient logging...for repairing a file system when its consistency is lost" during a system crash.

(Abstract). This is in direct contrast to Applicant's implementation of "dirty" flags to indicate whether "the control buffer data structure is marked identifying the file as being dirty for the current [or next] CP." (page 6, line 23-25).

Koseki uses "consistency" as defined by any differences (inconsistencies) between metadata originally stored in the metadata volume and any modified metadata objects that are eventually stored in the log volume. [col. 20, lines 16-60]. Conversely, Applicant defines a consistency point as "a wholly consistent and up-to-date version of the file system that is typically written to disk or to other persistent storage media...[where a consistency point] of the file system is [automatically] generated ...at regular time intervals." (page 4, lines 24-26). Furthermore, it should be noted that Koseki only checks and corrects "inconstancies in the metadata volume...in the case of an abnormal system shutdown." [col. 8, lines 37-52].

Applicant respectfully urges that the Koseki patent is legally precluded from anticipating the claimed invention under 35 U.S.C. § 102 because of the absence from the Koseki patent of Applicant's use of indicating a buffer data control structure as being "***dirty for a next consistency point.***"

Similarly, independent claims 21 and 27 include limitations of "current consistency point" and "next consistency point" as used in Applicant's invention. As noted above, Koseki does not teach or disclose the concept of a consistency point. As such, Koseki does not anticipate these claims.

Additionally, the present invention, as set forth in representative claim 15 comprises:

15. A storage system adapted to enable write operations to a file undergoing write allocation, the storage system comprising:

a write allocation process of a file system, the write allocation process adapted to associated received file data with a buffer data control structure upon receipt of a write operation directed to the file ***while the file is undergoing write allocation.***

Koseki teaches the use of a "log collection unit [collecting] log information, i.e., what modifications have been made to the metadata objects in the metadata cache. Such log records are accumulated in a log buffer, and a log writing unit transfers them from the log buffer to the log volume." Only "after an abnormal system shutdown...[will] a file system restoration unit [repair] the file system in response to a file system restoration request from an external source" and cannot write allocate to a file while that file is undergoing write allocation. [col. 7, lines 45 to col. 8, lines 61].

Applicant respectively urges that Koseki does not show Applicant's claimed novel ability to write allocate to a file "***while the file is undergoing write allocation.***"

As stated above, Koseki's uses the term "consistency" is defined by any differences (inconsistencies) between metadata originally stored in the metadata volume and any modified metadata objects that are eventually stored in the log volume. [col. 20, lines 16-60]. Conversely, Applicant defines consistency point as "a wholly consistent and up-to-date version of the file system that is typically written to disk or to other persistent storage media [where a consistency point] of the file system is [automatically] generated ...at regular time intervals." (page 4, lines 24-26). Koseki ***repairs*** discrepancies in metadata after an abnormal system shutdown by having the "file system restoration unit read out 'valid' log records from the log volume" and then comparing them to modified metadata stored in the metadata volume matched by a sequence number allocated by a sequence number assignment unit. (col. 8, lines 9-61; 30b of Fig. 15). In direct contrast, Applicant's invention is not restricted to abnormal system shutdown to perform any operation of Applicant's invention and "may generate [a CP] because, for example, a predetermined time period may have elapsed since the last CP or the amount of available memory in the storage system has been reduced to a level where the CP is needed to 'free up' memory space to allow continued buffering of write operations to a file ***while the file is [currently] undergoing write allocation.*** (page 21, lines 18-22). This is another distinction between Koseki and Applicant's claimed invention.

When the system described in Koseki is faced with more transaction requests than the system can handle, a "transaction limiter decides whether to accept or deny those requests, based on the current status of the logging system" i.e., "how much the log volume is occupied by valid log records." If the log volume is filled to or near a predetermined threshold, "the transaction limiter would not accept any new transactions...until space is "gradually decreased as the metadata writing unit writes updated metadata objects back to the metadata volume....the transaction limiter then resumes accepting new transactions [once again]. (col. 14, lines 10-50; 90a of Fig 9). The elimination of the need for transaction limiters like that in Koseki, that are caused by "a large number of incoming write while the CP write allocation operation is performed," is one of the operation of Applicant's invention. (page 5, lines 15-24; page 6, lines 8-25).

Applicant respectfully urges that the Koseki patent is legally precluded from anticipating the claimed invention under 35 U.S.C. § 102 because of the absence from the Koseki patent of Applicant's ability to write allocate to a file "***while the file is undergoing write allocation.***"

All independent claims are believed to be in condition for allowance.

All dependent claims are believed to be dependent from allowable independent claims, and therefore in condition for allowance.

Favorable action is respectfully solicited.

Should the Examiner feel personal contact is required to discuss this matter further, please do not hesitate to call the undersigned attorney at (617) 951-2500.

Please charge any additional fee occasioned by this paper to our Deposit Account No. 03-1237.

Respectfully submitted,

Duane Hay Dreger

Duane H. Dreger
Reg. No. 48,836
CESARI AND MCKENNA, LLP
88 Black Falcon Avenue
Boston, MA 02210-2414
(617) 951-2500